What is claimed is:

1.   A method comprising a first process for a first pixel and a second process for the first pixel, wherein:

the first process includes extracting a first kernel from a multi-color matrix, generating first variance weights from the first kernel, and generating a first color based on the first variance weights and adjacent pixel values of the first color; and

the second process includes extracting a second kernel from the multi-color matrix, generating second variance offsets from the second kernel, and generating a second color based on the second variance offsets and an adjacent pixel of the second color.

2.   The method of claim 1, further comprising a third process for a second pixel, wherein the third process includes extracting a third kernel from the multi-color matrix, generating third variance offsets from the third kernel, and generating a third color based on the third variance offsets and an adjacent pixel of the third color.

3.   The method of claim 1, wherein the first kernel is a 5x5 matrix and the second kernel is a 3x3 matrix.

4.   The method of claim 1, wherein generating the first variance weights includes determining horizontal and vertical gradient value averages.

5.   The method of claim 4, wherein:

the first kernel includes first and second side pixels of the first color;

determining the horizontal gradient value average includes determining a horizontal gradient value and calculating an average of the horizontal gradient value and at least one more horizontal gradient value; and

determining the horizontal gradient value includes calculating an absolute value of a difference between a value of the first side pixel and a value of the second side pixel.

6.   The method of claim 4, wherein:

15

the first variance weights include horizontal and vertical interpolation weights; and

generating the first variance weights further includes determining the horizontal and vertical interpolation weights based on the horizontal and vertical gradient value averages and a predetermined exponent value.

7.   The method of claim 6, wherein:

the horizontal interpolation weight is calculated based on

$$\lambda_H = 1 - \delta^k_H / (\delta^k_H + \delta^k_V); \text{ and}$$

the vertical interpolation weight is calculated based on

$$\lambda_V = 1 - \delta^k_V / (\delta^k_H + \delta^k_V),$$

where δ^k_H is the horizontal gradient value average, δ^k_V is the vertical gradient value average, and k is the predetermined exponent value.

8.   The method of claim 1, wherein generating the second variance offsets includes determining gradients between pixels of the first color and determining gradient values corresponding to the gradients.

9.   The method of claim 8, wherein generating a second color includes:

choosing a minimum value of the gradient values;

selecting the adjacent pixel of the second color based on a gradient that corresponds to the minimum value; and

subtracting the minimum value from a pixel value of the selected adjacent pixel.

10.   The method of claim 2, wherein the first kernel is a 5x5 matrix, the second kernel is a 3x3 matrix, and the third kernel is a 3x3 matrix.

11.   The method of claim 2, wherein generating the third variance offsets includes determining gradients between pixels of the first color and determining gradient values corresponding to the gradients.

12.   The method of claim 11 wherein generating a third color includes:

choosing a minimum value of the gradient values;

selecting the adjacent pixel of the third color based on a gradient that corresponds to the minimum value; and

subtracting the minimum value from a pixel value of the selected adjacent pixel.

13.   The method of claim 2, wherein the multi-color matrix is a Bayer matrix, and the first color is green.

14.   The method of claim 13, wherein the second color is blue and the third color is red.

15.   A computing machine comprising a processor and a memory, wherein:

the memory stores a first module for controlling the processor to perform a first process for a first pixel, wherein the first process includes extracting a first kernel from a multi-color matrix, generating first variance weights from the first kernel, and generating a first color based on the first variance weights and adjacent pixel values of the first color; and

the memory further stores a second module for controlling the processor to perform a second process for the first pixel, wherein the second process includes extracting a second kernel from the multi-color matrix, generating second variance offsets from the second kernel, and generating a second color based on the second variance offsets and an adjacent pixel of the second color.

16.   The machine of claim 15, wherein:

the memory further stores a third module for controlling the processor to perform a third process for a second pixel; and

the third process includes extracting a third kernel from the multi-color matrix, generating third variance offsets from the third kernel, and generating a third color based on the third variance offsets and an adjacent pixel of the third color.

17.   The machine of claim 15, wherein the first variance weights include horizontal and vertical gradient value averages.

18.   The machine of claim 17, wherein:

the first kernel includes first and second side pixels of the first color;

the first module includes logic to determine the horizontal gradient value average by determining a horizontal gradient value and calculating an average of the horizontal gradient value and at least one more horizontal gradient value; and

the first module further includes logic to determine the horizontal gradient value by calculating an absolute value of a difference between a value of the first side pixel and a value of the second side pixel.

19.   The machine of claim 17, wherein:

the first variance weights include horizontal and vertical interpolation weights; and

the first module includes logic to generate the first variance weights by determining the horizontal and vertical interpolation weights based on the horizontal and vertical gradient value averages and a predetermined exponent value.

20.   The machine of claim 19, wherein:

the horizontal interpolation weight is calculated based on

$$\lambda_H = 1 - \delta^k_H / (\delta^k_H + \delta^k_V); \text{ and}$$

the vertical interpolation weight is calculated based on

$$\lambda_V = 1 - \delta^k_V / (\delta^k_H + \delta^k_V),$$

where δ^k_H is the horizontal gradient value average, δ^k_V is the vertical gradient value average, and k is the predetermined exponent value.

21. The machine of claim 15, wherein the second module includes logic to generate the second variance offsets by determining gradients between pixels of the first color and determining gradient values corresponding to the gradients.

22. The machine of claim 21, wherein the second module includes logic to generate a second color by choosing a minimum value of the gradient values, selecting the adjacent pixel of the second color based on a gradient that corresponds to the minimum value, and subtracting the minimum value from a pixel value of the selected adjacent pixel.

23. The machine of claim 16, wherein the third module includes logic to generate the third variance offsets by determining gradients between pixels of the first color and determining gradient values corresponding to the gradients.

24. The machine of claim 23, wherein the third module includes further logic to generate a third color by choosing a minimum value of the gradient values, selecting the adjacent pixel of the third color based on a gradient that corresponds to the minimum value, and subtracting the minimum value from a pixel value of the selected adjacent pixel.

25. A computer readable medium having stored thereon a plurality of modules for controlling a processor, the plurality of modules comprising:

a first module for controlling the processor to perform a first process for a first pixel, wherein the first process includes extracting a first kernel from a multi-color matrix, generating first variance weights from the first kernel, and generating a first color based on the first variance weights and adjacent pixel values of the first color; and

a second module for controlling the processor to perform a second process for the first pixel, wherein the second process includes extracting a second kernel from the multi-color matrix, generating second variance offsets from the second kernel, and generating a second color based on the second variance offsets and an adjacent pixel of the second color.

19

26.   The media of claim 25, further comprising a third module for controlling the processor to perform a third process for a second pixel, wherein the third process includes extracting a third kernel from the multi-color matrix, generating third variance offsets from the third kernel, and generating a third color based on the third variance offsets and an adjacent pixel of the third color.

27.   The media of claim 25, wherein the first variance weights include horizontal and vertical gradient value averages.

28.   The media of claim 27, wherein:
        the first kernel includes first and second side pixels of the first color;
        the first module includes logic to determine the horizontal gradient value average by determining a horizontal gradient value and calculating an average of the horizontal gradient value and at least one more horizontal gradient value; and
        the first module further includes logic to determine the horizontal gradient value by calculating an absolute value of a difference between a value of the first side pixel and a value of the second side pixel.

29.   The media of claim 27, wherein:
        the first variance weights include horizontal and vertical interpolation weights; and
        the first module includes logic to generate the first variance weights by determining the horizontal and vertical interpolation weights based on the horizontal and vertical gradient value averages and a predetermined exponent value.

30.   The media of claim 29, wherein:
        the horizontal interpolation weight is calculated based on
$$\lambda_H = 1 - \delta^k_H / (\delta^k_H + \delta^k_V); \text{ and}$$
        the vertical interpolation weight is calculated based on
$$\lambda_V = 1 - \delta^k_V / (\delta^k_H + \delta^k_V),$$

where δ^k_H is the horizontal gradient value average, δ^k_V is the vertical gradient value average, and k is the predetermined exponent value.

31.   The media of claim 25, wherein the second module includes logic to generate the second variance offsets by determining gradients between pixels of the first color and determining gradient values corresponding to the gradients.

32.   The media of claim 31, wherein the second module includes logic to generate a second color by choosing a minimum value of the gradient values, selecting the adjacent pixel of the second color based on a gradient that corresponds to the minimum value, and subtracting the minimum value from a pixel value of the selected adjacent pixel.

33.   The media of claim 26, wherein the third module includes logic to generate the third variance offsets by determining gradients between pixels of the first color and determining gradient values corresponding to the gradients.

34.   The media of claim 33, wherein the third module includes further logic to generate a third color by choosing a minimum value of the gradient values, selecting the adjacent pixel of the third color based on a gradient that corresponds to the minimum value, and subtracting the minimum value from a pixel value of the selected adjacent pixel.

35.   A method for object-based color reconstruction in a multi-color matrix-based sensor arrangement comprising color sensors that have one first luminance component sensed at a relatively higher spatial frequency and two further chrominance components sensed at relatively lower spatial frequencies, for a particular pixel not sensed in said first luminance component estimating its first color value through determining local gradients among various first luminance component values assessed, said method being characterized by executing the following steps:

and in accordance with such local gradients executing such estimating through along relatively stronger edge informations, interpolating with a relatively greater weight

factor, in favor over interpolating along relatively weaker edge informations with a relatively lesser weight factor,

and for a particular pixel not sensed in a particular further chrominance component value estimating that further chrominance component's value in a direction along with relatively smaller differences evaluated in said first luminance component.

36.   A method as claimed in Claim 35, wherein further chrominance component's values of said particular pixel are interpolated using neighboring pixels' information as based on whether they are situated within a same imaged object.

37.   A method as claimed in Claim 35 and applied to a Bayer matrix wherein said first luminance component is green.

38.   A method as claimed in Claim 36, wherein said first luminance component's value is estimated on the basis of a 5 x 5 pixel kernel centered on said particular point.

39.   A method as claimed in Claim 35, whilst through exponential gradient values adjusting an exponent value (k) for emphasizing edge clarity in a low-noise situation, or rather limiting noise propagation whilst still mitigating for color aliasing.

40.   A method as claimed in Claim 35, whilst adding a low-pass filter step after estimating non-sampled colors for mitigating false-color spikes.

41.   A method as claimed in Claim 40, whilst supplementing said low pass filter step with relatively enhancing spatial high frequencies relatively far from said low-pass filter's discriminatory frequency for edge sharpness enhancement (Figure 5).

42.   A computer program comprising program instructions for controlling a computer to implement a method according to one of Claims 35 to 41.

43.   A computer program product as being represented within a tangible read-only computer memory medium or being carried by an electrical signal, and comprising program instructions for controlling a computer to implement a method according to one of Claims 35 to 41.

44.   An apparatus being arranged for implementing a method as claimed in Claim 35.

45.   An apparatus according to Claim 44, and executed as a filter facility for limiting noise propagation.

46.   An image sensing facility comprising an image forming facility for forming an image on an apparatus as claimed in Claims 44 or 45, and furthermore comprising a memory facility fed by said apparatus, processing means for dynamically interacting with pixel values in said memory and user output means for outputting a reconstructed user image.